SPRINT22

BoWとは何だったか

BoWによる表現（TF-IDFを含む）は、文書と単語からなる行列でした。そして、一つの次元が一つの単語に対応するという関係から、それは巨大な疎行列になりかねません。
scikit-learnの「Feature extraction /4.2.3.2. Sparsity」ではこれを以下のように評しています。

「ほとんどの文書においては、コーパス（BoWの列のこと）のとても小さな単語集合を扱っているため、結果として得られるBoWの99%以上が0になってしまいます。例として10,000の短文の文書群を考えた際に、100,000の単語によって構成される一方で、一つの文書あたりで使用する重複のないユニークな単語の数は100~1,000ほどです。この疎行列を扱うためにscipy.sparseのようなsparse representationを用います。」

https://scikit-learn.org/0.16/modules/feature_extraction.html#sparsity

局所表現と分散表現

前述のBoW的な表現は、**局所表現**（local representation）と呼ばれます。

これに対し、Word2Vecによって生み出される表現は**分散表現**（distributed representation）と呼ばれ、複数の次元が一つの単語を構成し、また一つの次元が複数の単語を構成するために用いられます。

Word2Vec = Word to Vector

Word2Vecは、どのようにして分散表現を作るのでしょうか。

まず、入力に100,000ほどのOne-hot 表現（局所表現）を用い、200ほどの隠れ層を通して次の単語を予測するモデルを作り、ある程度うまく予測できるようになったとします。
このとき、100,000個の単語の情報は、200個の隠れ層つまり次元圧縮された潜在変数の空間にマッピングされたと捉えることができます。言い換えると、ここの200個のパラメータは、100,000個の単語の情報を保持していると考えることができます。

ゆえに、この200個のパラメータを分散表現として用いようというのがWord2Vecのアイデアです。

それでは、この分散表現を深層学習の言語モデルの入力データとしましょう。

深層学習の言語モデル：RNNとは？

Recurrent Neural Networks

自然言語処理分野では、連続データの分析に適した深層学習モデルとして、RNN（Recurrent Neural Networks）系のモデル（RNNとその系譜）が現在も用いられています。

Recurrentは「循環する」と言う意味を持ちます。

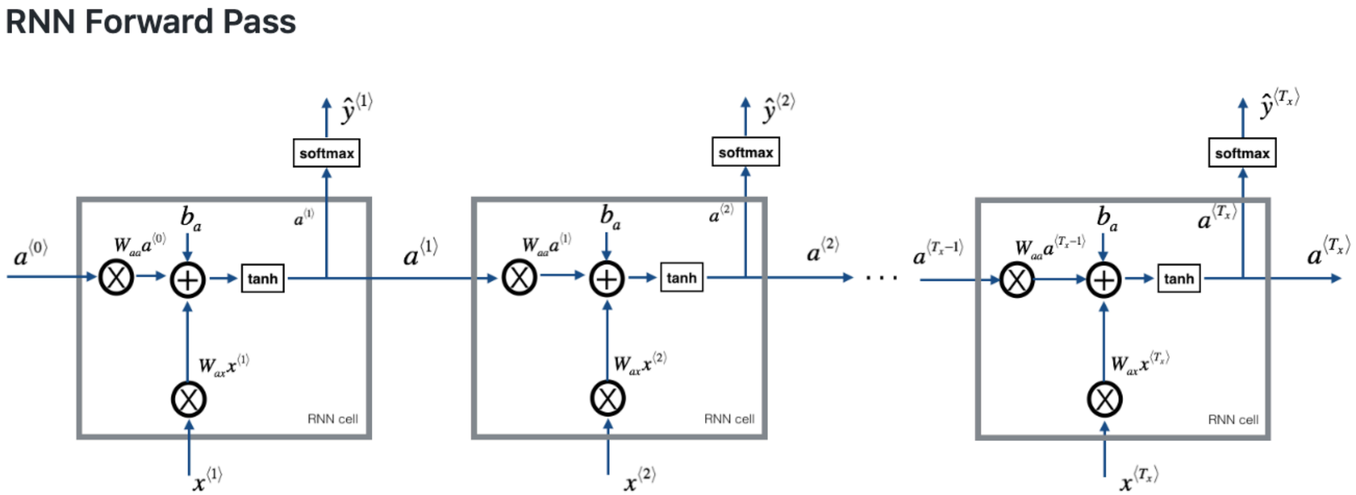
「循環する」ためにループする経路を持ち、そのループにおいて過去の情報を記憶しつつ、それをまた新しい情報へ更新しながら保持していきます。

RNNのネットワーク

RNNのフォワードは右のように表現される
ネットワークです。

四角い範囲から四角い範囲へ
なにかがパスされています。

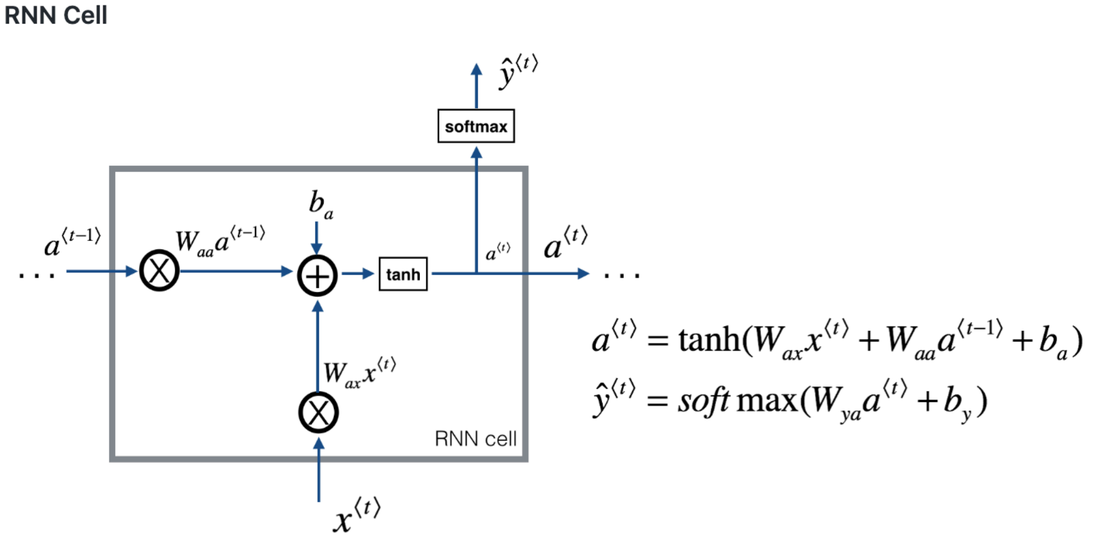
どこで循環的しているのだろう。



RNNのネットワーク

四角い範囲の並びは、同一のネットワーク（セルと呼ぶ）を時間軸方向に展開したものです。
同じネットワークで演算された出力（a (t-1)）を自分の中へ入力（a (t)）し、また演算することを反復します。
これが再帰的なネットワークと言われる所以です。

ふたつの重み、W_axとW_aaは、それぞれの時間において共有されます。



RNNのネットワーク